A Real-Time Street Actions Detection

Salah Alghyaline
Department of Computer Science
The World Islamic Sciences and Education University, Amman, Jordan

Abstract—**Human action detection in real time is one of the most important and challenging problems in computer vision. Nowadays, CCTV cameras exist everywhere in our lives. However, the contents of these cameras are monitored and analyzed using human operator. This paper proposes a real time human action detection approach which efficiently detects basic and common actions in the street such as stopping, walking, running, group stopping, group walking, and group running. The proposed approach measures the object movement type based on three techniques: YOLO object detection, Kalman Filter and Homography. Real videos from CCTV camera and BEHAVE dataset are used to test the proposed method. The experimental results show that the proposed method is very effective and accurate to detect basic human actions in the street. The accuracies of the proposed method on the tested videos are 96.9% and 88.4% for the BEHAVE and the created CCTV datasets, respectively. The proposed approach runs in real time with more than 50 fps for BEHAVE dataset and 32 fps for the created CCTV datasets.**

Keywords—Online human action detection; group behavior analysis; CCTV cameras; computer vision

I. Introduction

Online human action recognition is a very challenging and unsolved problem in computer vision. The aim of action recognition is to recognize human action in a streaming video or a live camera as soon as possible or even before the action is completed. Human action recognition has many applications such as visual surveillance, video content analysis, and human-computer interaction. Nowadays, we have millions of CCTV cameras everywhere, and human operators are used to monitor the output. However, using humans for monitoring CCTV camera is very expensive and unreliable way to check the CCTV contents. Therefore, it is crucial to develop an automated way for analyzing the content of CCTV cameras. There are many limitations for using the current approaches and datasets in action recognition [1]. In the existing action recognition datasets, each video clip contains a single action type from the beginning to the end of the clip, therefore it is necessary to determine the duration of the action in the video. Whereas in CCTV videos, the action could occur at any time and many action types usually occur in the same scene. Moreover, most of these datasets have a limited number of actions but in the real world there are many actions. Some of these datasets were created especially for testing purposes. In these datasets, video actions are not real and are captured under specific conditions of lighting occlusion and clutter to make them clearer and visible for the testing stage. In the existing action detection methods, most of the existing action recognition approaches [2] [3] work offline and are based on the Bag-of-words (BoW) model [4]. In BoW model, processing one video clip passes through many independent time-consuming stages; it starts by detecting the interest points for each frame, then tracking these interest points in a sequence of frames, after that describing the interest points spatially and temporally using a descriptor such as HOG (Histograms of Oriented Gradients) [5], HOF (Histograms of Optical Flow) [6], SIFT (Scale-invariant feature transform) [7] and SURF (Speeded-up Robust-Features) [8], then clustering the features into a specific number of visual words, this step is usually done using k-means. The visual words are then used to represent each video by a histogram of visual words. Finally, support vector machine (SVM) is used to classify the videos into different kind of actions. In addition to the time consumption problem, the hand-crafted features (such as HOG, HOF, SIFT…) lack the ability to find semantic or meaningful features that can discriminate the action type accurately. Currently Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have been used in action recognition to achieve superior results [9] [10] [11]. Compared with the traditional hand-crafted based approaches, a deep neural network is employed to automatically discover the semantic features from a large group of videos, however, the majority of the proposed approaches in action recognition are based on CNNs and RNNs and designed for offline detection, and there are few works done in online action detection [12] [13].

To address the above issues, this paper proposes a novel approach to analyze human actions in real time, which makes it applicable for CCTV camera. The proposed action detection method is based on You only Look Once (YOLO) the-state-of the-art in object detection, Kalman filter and Homography. Basically, YOLO is used to detect the required objects and their types inside a single frame, after that the detected objects are tracked along these frames using Kalman filter, then extracting the trajectory of the moving object, finally Homography is used to determine the movement type based on the moved distance during a specific duration of time. In the real world, each surveillance system is interested in a specific kind of actions that serve the business needs, therefore the proposed approach focuses on a specific kind of behaviors. Additionally, a dataset using real live CCTV videos are created to test the performance of the proposed method, unlike many existing methods in action recognition that use short clips or videos that are captured under some circumstances to reduce the noise during the detection process.

The contributions of this paper can be summarized as follows:

- The paper focuses on explaining and finding solutions for the online human action detection problem.

- The paper develops action detection system based on three well known approaches in computer vision. YOLO object detection, Kalman filter approach and Homography.
- Building dataset that includes long and real video streams for online action detection problem. The videos duration is 4 hours and 11 minutes, the dataset videos were captured from live CCTV camera and can be used for training and testing purposes.

II. Related Works

A. Action Recognition

Action recognition is the ability to detect the action type from a movable object. In general, action recognition is used to analyze human behaviors through surveillance systems, and RGB camera is used to get the input data. Human action recognition attracted many researchers during the last few years for security concerns, however, it is very challenging to develop accurate and real-time applications to recognize human actions automatically from real world scene. Mainly, there are two kinds of features that are used for action recognition: hand-crafted features (like HOG, HOF, and MBH, etc.) and deep learning features (based on convolutional neural networks).

B. Group Action Recognition

In action recognition the action is performed by a single person, two persons (interaction), or by large number of persons (Crowed), or by two to view number of persons (group action recognition). Cho et al. [14] proposed a method to address the group action recognition problem, the approach proposes to use Group Interaction Zone (GIZ), and the interaction between people is classified into four categories: intimate, personal, social and public, this classification is based on proxemics. Attraction and repulsion concept are used to describe the action type, where the object is moving closer or a way from each other.

Yin et al. [15] proposed a framework for small group action recognition, the approach has four stages: mean-shift tracker is used to track the object position during its movement, then clustering the object coordinates into a number of groups, after that building a descriptor based on social network analysis features . Finally, a Gaussian model is used to model different action types. However, most of the proposed approaches in group action recognition are not real time approaches, moreover they do not implement object detection phase and use Ground truth information to know the exact object location (they assume that the object locations are known before the recognition process).

III. Proposed Online Action Recognition Method

As it is shown in Fig. 1 there are four stages for recognizing the action type in real time according to the proposed method: detecting the object, tracking the object, extracting the movement trajectory and using the Homography to make the recognition decision.

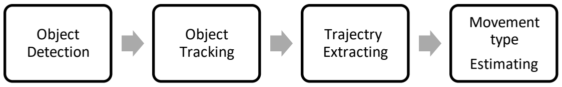


Fig. 1. Main Steps of the Proposed Action Detection System.

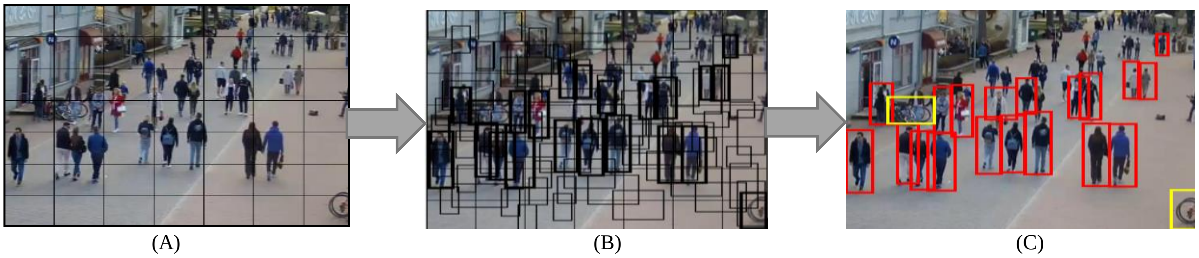
A. Object Detection

There are many proposed algorithms to solve the problem of object detection with high accuracy. Faster R-CNN is an object detection algorithm based on convolution neural network. It can detect the object, give the probability score for the detection, and predict bounding box position at the same time. This algorithm is different from the previous CNN object detection algorithms (Fast R-CNN, Spatial pyramid pooling in deep convolutional networks for visual Recognition), it does not consume additional time for region proposal because it uses shared convolutional network for identifying the object type and the object position at the same time. Faster R-CNN has a good detection accuracy compared with the existing approaches, however the detection speed is about 5 fps which makes it difficult to be used with real time applications. You Only Look Once (YOLO9000) is the-state-of-the-art real time object detection; it is reported that it has slightly better accuracy compared with Faster R-CNN algorithm, and it is much faster (67 fps). Additionally, it can detect objects in a higher speed than real time. One neural network evaluation is used for making prediction for one image which saves a lot of time compared with other R-CNN approaches.

In object detection stage a CNN model is trained based on YOLO architecture. 1600 pictures were captured at different times during the years 2017 and 2018 (summer, winter, morning and evening) from Baltic Live Cam[1], it is a live streaming camera broadcasts live images from Jomas Street in Jurmala one of the famous cities in Latvian. It has been noticed that there are three kinds of objects moving in that street: persons, bikes and strollers, therefore the number of classes was set to 3 during the training stage. We stopped learning the model after 60700 iterations and the average loss value was close to 0.6. There was no significant reduction of loss value after 60700 iterations. BEHAVE dataset is smaller compared with the created CCTV dataset. A sample frames from BEHAVE dataset are also used to train YOLO model.

The input image passes through 19 convolutional neural network layers and 5 max pooling layers, followed by average pooling layer and finally soft max layer. In Fig. 2 (a), the input image with the dimensions 416×416 is divsided into equal sizes of $S \times \mathrm{S}$ grid, the final output after applying a sequence of convolutions and pooling layers will be a feature map with the size 13×13 (similar to the number of grids). The final number of tensors for each image is $13 \times 13 \times 40$, where 13×13 denotes the number of grids and $40(5 \times 8)$ is 5 bounding boxes each box has 8 floating point numbers, the 8 numbers as follows: 3 is the probability for each class (person, bike and stroller) and 5 numbers for the bounding box (x, y) coordination, width, height of the rectangular box and object confidence score.

[1] "https://balticlivecam.com/cameras/latvia/jurmala/cafe-3/," Baltic Live Cam, 2018. [Online].



(A) (B) (C)

Fig. 2. YOLO Object Detection Steps: A) Split the Image Into S×S Grid, B) Predict the Bounding Boxes and the Confidence of Each Box C) Make Final Prediction.

B. *Object Tracking*

Object tracking is one of the challenging problems in computer vision due to different reasons such as: object detection, occlusions and sudden movement in object location. Kalman filter (KF) with Hungarian algorithms [16] is one of the most used methods for object tracking. The following books and papers [17] [18] [19] describe in detail how does KF work. In KF, the new location of the object can be predicted based on the object movement model and the measured location. At the beginning, KF algorithm predicts the current object location based on the previous position and the motion model (e.g. Motion laws), the prediction probability of this predication is also calculated. In the next step, the measured location of the moving object is obtained (YOLO is used for detecting the object location), the final step is to update the final estimated position by giving a weight for measured and predicted positions, this weight is called Kalman gain. If the gain value is low, then the estimated position tends to be close to predicted value, whereas if the gain value is high, then the estimated position is following the measured value. In many cases, YOLO is not able to detect some of the objects in the frame, therefore KF will not be able to get the measured object position, and in this case the prediction process is used to calculate object location without performing the update state. Different equations of KF will be explained below.

$$\hat{y}^-_k = Ay_{k-1} + Bu_k \tag{1}$$

The projection of new state is shown in Eq. (1), where $\hat{y}^-_k$ denotes the state of the system at time *k*. *A* is the system model that predicts the new location of the object (the model is based on motion laws) and y_{k-1} is the previous location of the object. *B* and u_k are the control model and control vector, respectively.

$$P^-_k = AP_{k-1}A^T + Q \tag{2}$$

The projection of error covariance is shown in Eq. (2), where *A* and A^T are the system model as before in Eq. (1) and the transposed of the model vector, respectively. P_{k-1} is the value of the error at time *k-1* and *Q* is the covariance of the noise error, which describes noise distribution.

$$K = P^-_kH^T(HP^-_kH^T + R)^{-1} \tag{3}$$

Eq. (3) shows the Kalman Gain equation, P^-_k is the covariance of the predicted error, *H is* the model of measurement, and *R* is the covariance of the measurement noise.

$$\hat{y}_k = \hat{y}^-_k + K(z_k - H\,\hat{y}^-_k) \tag{4}$$

Eq. (4) explains updating the estimation which gives the final output of the KF. $\hat{y}_k$ is the output of Kalman filter and it describes the object state at time *k*, $\hat{y}^-_k$ is the previous object state, *K* denotes the Kalman Gain, z_k is the measured value and $H\,\hat{y}^-_k$ *is* the predicted measurement.

$$P_k = (I - KH)P^-_k \tag{5}$$

Updating the error covariance is shown in Eq. (5), where *I* is the identity matrix, *K* represents the Kalman Gain, *H is* the model of measurement and P^-_k is previous error covariance.

In muli-object tracking it is necessary to apply optimal assignment between the detected objects in the current frame *F* and the previous frame *F-1*. First, the distance between the object locations is calculated using Eq. (6), then the Hungarian algorithm is used to make the mapping between the detected objects in frames *F* and *F-1*

$$D(p,q) = \sqrt{(x_2-x_1)^2+(y_2-y_1)^2}, p = (x_1,y_1), q = (x_2,y_2) \tag{6}$$

C. *Action Recognition*

Detecting and tracking the objects are important stages to identify the type of the action. The detection stage identifies the object type, whereas the tracking stage recognizes the movement type based on the trajectory of the moving object. Before making this project, it has been recognized that there are mainly three kinds of moving objects in the selected street (Jomas Street - Jurmala - Latvian): persons, strollers and bikes. Moreover, there are three kinds of movements: Walking, running and stopping actions. The previous objects types and actions can be performed by a single or a group of objects, so we have another three new action types Group walking, group running and group stopping actions.

By using object tracker, we can get the object locations during its movement from one point to another in 2D plane. However, the coordinates in the image are measured by pixels, whereas in the real world the distance between different objects are measured by centimeter or meter. Homography H is used to make the projection between 2D image coordinates and 3D real world

In Eq. (7), $(u\ v\ 1)^T$represents a 3D real world point in homogenous coordinate, $(x\ y\ 1)^T$represents a 2D image coordination, and H is the Homography matrix. The Homography is calculated according to this formula $AH = 0$, where A is $2n \times 9$ matrix, and n is the number of used points to find the Homography.

$$\begin{pmatrix}u\\v\\1\end{pmatrix} = H\begin{pmatrix}x\\y\\1\end{pmatrix} \tag{7}$$

$$H = \begin{pmatrix}h1 & h2 & h3\\h4 & h5 & h6\\h7 & h8 & h9\end{pmatrix} \tag{8}$$

$$A = \begin{pmatrix}x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1X_1 & -y_1X_1 & -X_1\\0 & 0 & 0 & x_1 & y_1 & 1 & -x_1Y_1 & -y_1Y_1 & -Y_1\\x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2X_2 & -y_2X_2 & -X_2\\0 & 0 & 0 & x_2 & y_2 & 1 & -x_2Y_2 & -y_2Y_2 & -Y_2\\\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\x_n & y_n & 1 & 0 & 0 & 0 & -x_nX_n & -y_nX_n & -X_n\\0 & 0 & 0 & x_n & y_n & 1 & -x_nY_n & -y_nY_n & -Y_n\end{pmatrix} \tag{9}$$

To distinguish between the three actions: Running, Walking and stopping we calculate the movement speed at a specific period of time T_thre, which means finding the walked distance D during T_thre. After that three thresholds are set for each kind of movements separately D_run, D_{walk} and D_stop. Finally, if $D\_walk \leq D < D\_run$, then the movement is classified as walking action. Similarly, the formula can be applied for other actions. In our experiment we set D_run, D_walk, D_stop to 5 m, 1.5 m and 0 m , respectively as it shown in Table 1, and T_thre is set to 3 seconds. The values for these thresholds are set based on experiments and showed a good result to discriminate these action types. The threshold gives us the upper bound distance, for example the walking distance will not exceed D_run (5 m). In the real situation the walked distance will be far from the threshold values for different actions, for example running action speed will be more than 7m in 3 seconds for the most running cases, and stopping distance is usually less than 1 m for the most stopping cases. However another threshold D_Group is used to define that a group of people is performing the action within one group, the group area is defined as a circle and the diameter of this circle is set to D_Group. D_Group is set to 3 meters in all experiments.

TABLE I. DISTANCE RANGE FOR STOPPING, WALKING AND RUNNING ACTIONS

Distance	Possible action type
0-1.5	Stopping
1.5-5	Walking
Above 5	Running

IV. DATASET

A. The Created Dataset from CCTV Videos

Unlike other datasets which are created under certain conditions of lighting, occlusion and clutter (to avoid noise during testing the videos) our dataset videos were captured from live camera, which makes the proposed system more effective and applicable for the real-world applications, the videos used in the experiments were captured during the years 2017 and 2018 in different seasons of the year. Table 2 shows the general characteristics of the used videos for testing the proposed approach. Mainly there are 8 continues videos were taken from Baltic Live Cam. The durations of these videos are ranged from 16 minutes to one hour and 40 minutes. The total time of all these videos is 251 minutes (4 hours and 11 minutes). The tested videos are 1920 pixels wide and 1080 pixels in height, whereas the frame rate is 30 frames per second (FPS) for 6 videos and 20 FPS for 2 videos. Fig. 3 shows sample pictures from the created CCTV dataset.

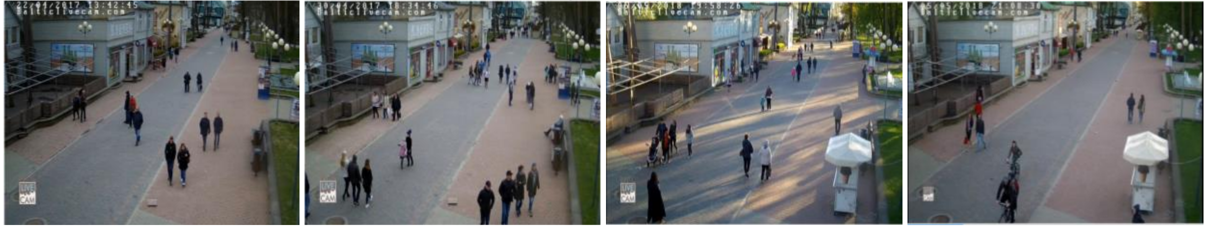


Fig. 3. Sample Pictures from the Created Dataset.

TABLE II. CHARACTERISTIC OF THE CREATED DATASET FOR TESTING THE PROPOSED METHOD

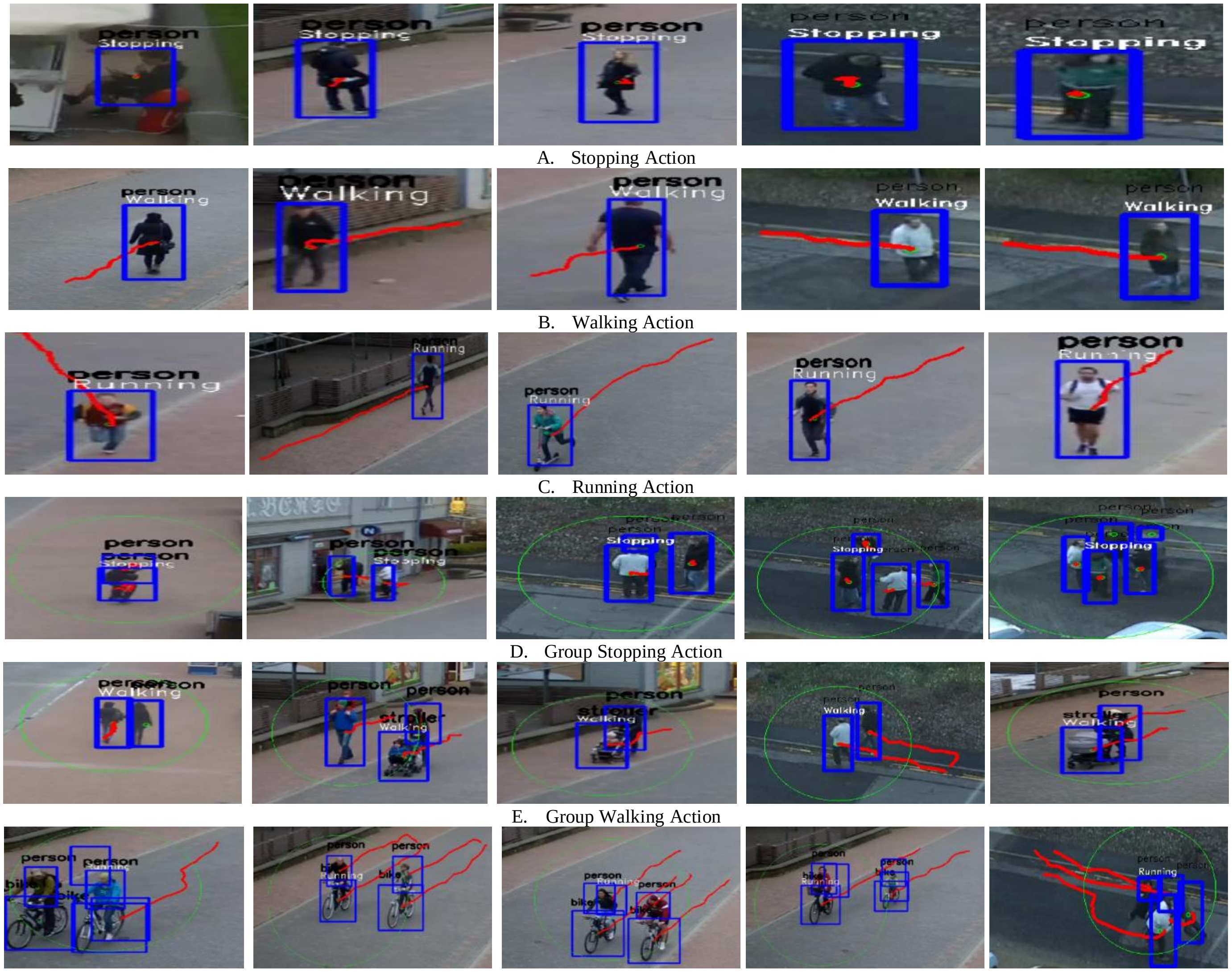
	Duration (minutes)	Format	Resolution(pixels)	Frame rate(frames/second)
Video 1	99:00	mp4	1920×1080	30
Video 2	9:58	mp4	1920×1080	20
Video 3	20:25	mp4	1920×1080	30
Video 4	16:29	mp4	1920×1080	30
Video 5	30:00	mp4	1920×1080	30
Video 6	35:30	mp4	1920×1080	30
Video 7	17:39	mp4	1920×1080	30
Video 8	22:34	mp4	1920×1080	20

B. BEHAVE Dataset

To show the efficiency of the proposed method a comparison with other approaches in action recognition is made on the BEHAVE dataset. Unlike other approaches, the proposed approach makes detection on the videos directly without using the Ground truth information that are provided by the BEAHAVE dataset. The dataset is used for group of people activities analysis. BEHAVE provides 10 classes of group activities, mainly: InGroup, WalkTogether, RunTogether, Approach, Meet, Ignore, Split, Fight, Chase, Following. The BEHAVE dataset includes 163 instances of these activities. The dataset is used for group behavior analysis therefor it does not count the individual activates instances (The activities that are done by single person) such as: walking, stopping and running. The frame resolution is 640×480, and the video rate is 25 fps.

V. Experiments

The proposed approach is implemented using C language on Intel (R) Core (TM) i5-8600k CPU @ 3.60GHz with 8 GB RAM and a NVIDIA GeForce GTX 1080 GPU. It is clear from the detection results Fig. 4 that the proposed action detection system is very effective and accurate to identify the 6 targeted action types Fig. 4(A) shows the results of detection stopping action type, a blue rectangle is surrounding the moving object, the action type and the moving object type are written above the box. Stopping action means that the object is not moving a lot and staying at the same place and that can be identified easily by the proposed approach by calculating the total movement during a specific period of time. As it is mentioned before the total object movement during the last T_thre seconds is calculated (T_thre is set to 3 seconds) and based on that distance the movement can be classified as stopping, walking and running action types. The red line behind the moving object represents the tracking path of the moving object, the green circle indicates that the action is performed with other objects like person or stroller, the proposed system also can identify the action type if it occurs in a group, according to the proposed system the moving objects are in one group if their locations are within one circle and the diameter of this circle is less than D_Group, D_Group is set to 5 meters in the experiments.



A. Stopping Action

B. Walking Action

C. Running Action

D. Group Stopping Action

E. Group Walking Action

F Group Running Action

Fig. 4. Samples of Detection Results using the Proposed Action Detection System for Each Action Type Separately.

It is not popular to see many running action types yet some of them were detected as shown in Fig. 4(C). Most of the group running actions were done by a group of persons that were riding bicycles as it shown in Fig. 4(F). Table 3 and Table 4 show the precision and the recall results for the proposed approach on the created CCTV dataset. The approach achieved high results especially for walking and group walking actions because in the walking action the object is moving forward with a regular speed and from the beginning to the end of the street which makes the possibility to detect and track the object easier. Another reason is that when the object is moving in a regular way, the object position, size and pose will be changed many times, this also makes the detection process for that object easier. Also, when a group of persons are waking together, the detection of this action will be higher, when there are six persons walking on the street at least four of them will be detected and tracked. It is clear that some videos do not have running actions since walking and group walking actions (families and friends) are the most common behaviors in that street. From Table 5, we can see that the precision for the created CCTV dataset is 92%, whereas the recall is 88.4%. Finally, the proposed approach can run in real time, it can process 32 frames per second; this time includes reading the video and output the detection results to the user on the screen.

Table 6 shows the confusion matrix for the proposed action recognition on BEHAVE dataset. Six action types were tested on this dataset: Stopping (S), Walking (W), Running (R), Group Stopping (GS), Group Walking (GW) and finally Group Running (GR). It is clear that the proposed system achieved high accuracy on this dataset for most of the six target actions. However, there is small percentage of confusion between some actions, for example 4.55% of walking actions were recognized as walking in a group, and 7.79% of walking in a group cases recognized as walking actions. This confusion, however, is related to the object detection accuracy, for example the YOLO approach could miss some objects on the scene or could make some false positive detections. For running actions, only four cases were noticed during the whole dataset, the overall accuracy for the proposed system after excluding the running action accuracy is 96.94%.

TABLE III. PRECISION FOR THE PROPOSED METHOD

	Stopping	**Walking**	**Running**	**StoppingInGroup**	**WalkingInGroup**	**RunningInGroup**
Video 1	82%	100%	100%	85%	100%	84%
Video 2	-	100%	-	-	100%	-
Video 3	100%	97%	-	75%	100%	100%
Video 4	100%	100%	-	86%	100%	100%
Video 5	91%	100%	-	63%	96%	92%
Video 6	81%	100%	100%	77%	99%	55%
Video 7	63%	100%	-	100%	99%	100%
Video 8	67%	100%	100%	71%	99%	100%
Average	83.4%	99.6%	100%	79.6%	99.1%	90.1%

TABLE IV. RECALL FOR THE PROPOSED METHOD

	Stopping	**Walking**	**Running**	**StoppingInGroup**	**WalkingInGroup**	**RunningInGroup**
Video 1	88%	97%	100%	92%	96%	100%
Video 2	-	100%	-	-	95%	-
Video 3	92%	100%	-	100%	96%	82%
Video 4	90%	100%	-	75%	97%	78%
Video 5	83%	94%	-	83%	96%	92%
Video 6	100%	98%	70%	81%	94%	86%
Video 7	83%	97%	-	80%	99%	75%
Video 8	100%	83%	67%	77%	88%	100%
Average	91.3%	96.1%	79%	81.3%	95.1%	87.6%

TABLE V. RECALL AND PRECISION OVERALL ACTIONS AND TESTED VIDEOS USING THE PROPOSED METHOD

	Precision	**Recall**
Average of all actions	92%	88.4%

TABLE VI. CONFUSION MATRIX OF THE PROPOSED ACTION RECOGNITION SYSTEM ON BEHAVE DATASET

	S	W	R	GS	GW	GR
S	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%
W	0.00%	95.45%	0.00%	0.00%	4.55%	0.00%
R	0.00%	0.00%	75.00%	0.00%	0.00%	25.00%
GS	2.94%	0.00%	0.00%	97.06%	0.00%	0.00%
GW	0.00%	7.79%	0.00%	0.00%	92.21%	0.00%
GR	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%

Comparisons with other human group behavior recognition approaches are made in Table 7 and Table 8. The advantages of the proposed method compared with these approaches are summarized on Table 8. The proposed method can run in more than the real time (from reading the videos frames to making the recognition decision). Another advantage is that there is no need to provide the system with the bounding boxes locations and the class of the object (Ground truth information). The proposed system can detect most of the needed objects and their locations with high accuracy. The proposed method also achieved the highest accuracy for group walking action types with accuracy 92.21%, and the average accuracy for the two compared actions; the Group stopping and Group walking is 94.64%. These accuracies indicate that the system is very efficient to detect the target actions even though it did not use the Ground truth information compared with other approaches and it can run in more than real time.

TABLE VII. PERFORMANCE COMPARISON WITH OTHER GROUP BEHAVIOR RECOGNITION APPROACHES

	The proposed approach	**Ref.** [14]	**Ref.** [20]	**Ref.** [15]	**Ref.** [21]
SG	97.06	100	90	94.3	88
WG	92.21	91.66	45	92.1	88
Average	94.64	95.83	67.5	93.2	88

TABLE VIII. THE ADVANTAGES OF USING THE PROPOSED METHOD COMPARED WITH OTHER EXISTING GROUP BEHAVIOR RECOGNITION APPROACHES

	The proposed approach	**Ref.** [14]	**Ref.** [20]	**Ref.** [15]	**Ref.** [21]
Run in Real time	Yes	No	No	No	No
Use Ground truth information	No	Yes	Yes	Yes	Yes

VI. CONCLUSION

This paper proposes a human action detection approach that can be used in the real time with live CCTV camera. The proposed approach is implemented based on three techniques: YOLO object detection which represents the state-of-the-art in object detection, Kalman filter which is one of the most successful techniques for object tracking and Homography to measure the object movement in meter. Another contribution in this paper is that it builds a dataset from a real live CCTV videos, the duration of these videos is more than four hours length, and they were taken under different conditions of lighting, clutter, scaling and occlusion. The experimental results on two datasets show that the proposed approach is very effective and accurate to detect most of the target actions in the tested videos, especially the most common actions in the street like: stopping, walking and running. Moreover, it can detect if the action is performed by a group of people or just by a single person. In future work, I will extend the number of detected action types.

REFERENCES

[1] R. De Geest, E. Gavves, A. Ghodrati, Z. Li, C. Snoek and T. Tuytelaars, “Online action detection,” in European Conference on Computer Vision, Springer, 2016, pp. 269--284.

[2] S. Alghyaline, J.-W. Hsieh, H.-F. Chiang and R.-Y. Lin, “Action classification using data mining and Paris of SURF-based trajectories,” in IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2016.

[3] S. Alghyaline, J.-W. Hsieh and C.-H. Chuang, “Video action classification using symmelets and deep learning,” in IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2017.

[4] L. Fei-Fei and P. Perona, “A Bayesian Hierarchical Model for Learning Natural Scene Categories,” in Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington, DC, USA, 2005.

[5] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2005.

[6] N. Dalal, B. Triggs and C. Schmid, “Human detection using oriented histograms of flow and appearance,” in European conference on Computer Vision (ECCV), Graz,Austria, 2006.

[7] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” International journal of computer vision, vol. 60, no. 2, pp. 91-110, 2004.

[8] H. Bay, A. Ess, T. Tuytelaars and L. Van Gool, “Speeded-Up Robust Features (SURF),” Computer Vision and Image Understanding, vol. 110, no. 3, pp. 346-359, 2008.

[9] H. Rahmani, A. Mian and M. Shah, “Learning a deep model for human action recognition from novel viewpoints,” IEEE transactions on pattern analysis and machine intelligence, vol. 40, no. 3, pp. 667-681, 2018.

[10] R. Hou, C. Chen and M. Shah, “Tube convolutional neural network (T-CNN) for action detection in videos,” in IEEE International Conference on Computer Vision, 2017.

[11] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar and L. Fei-Fei, “Large-scale video classification with convolutional neural networks,” in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2014.

[12] S. Baek, K. I. Kim and T.-K. Kim, “Real-time online action detection forests using spatio-temporal contexts,” in IEEE Winter Conference on Applications of Computer Vision (WACV), 2017.

[13] J. Liu, Y. Li, S. Song, J. Xing, C. Lan and W. Zeng, “Multi-Modality Multi-Task Recurrent Neural Network for Online Action Detection,” IEEE Transactions on Circuits and Systems for Video Technology, 2018.

[14] N.-G. Cho, Y.-J. Kim, U. Park, J.-S. Park and S.-W. Lee, “Group activity recognition with group interaction zone based on relative distance between human objects,” International Journal of Pattern Recognition and Artificial Intelligen, vol. 29, no. 5, p. 1555007, 2015.

[15] Y. Yin, G. Yang, J. Xu and H. Man, “Small group human activity recognition,” in 19th IEEE International Conference on Image Processing (ICIP), 2012.

[16] H. W. Kuhn, “The Hungarian method for the assignment problem,” Naval research logistics quarterly, vol. 1, pp. 1-2, 1955.

[17] S. Bozic, Digital and Kalman filtering: an introduction to discrete-time filtering and optimum linear, New York, NY: Halsted Press, 1994.

[18] R. G. Brown and P. Y. Hwang, Introduction to random signals and applied Kalman filtering, New York: Wiley, 1992.

[19] G. Welch and G. Bishop, “An introduction to the Kalman filter,” in Proc of SIGGRAPH, Course, 8(27599-3175), 59., 2001.

[20] D. Münch, E. Michaelsen and M. Arens, “Supporting fuzzy metric temporal logic based situation recognition by mean shift clustering,” in Annual Conference on Artificial Intelligence, Berlin, Heidelberg, 2012.

[21] C. Zhang, X. Yang, W. Lin and J. Zhu, “Recognizing human group behaviors with multi-group causalities,” in Proceedings of the The 2012 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology, Macau, China, 2012.